# UNITED STATES PATENT AND TRADEMARK OFFICE

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/620,747 | 07/16/2003 | Mark S. Moir | 6000-33600 | 8970 |

58467      7590      08/06/2009
MHKKG/SUN
P.O. BOX 398
AUSTIN, TX 78767

| EXAMINER |
|---|
| TECKLU, ISAAC TUKU |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2192 | |

| NOTIFICATION DATE | DELIVERY MODE |
|---|---|
| 08/06/2009 | ELECTRONIC |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

patent_docketing@intprop.com
ptomhkkg@gmail.com

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *08 April 2009*.

2a)☒ This action is **FINAL**.          2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-43 and 46-59* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-43 and 46-59* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All   b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
    Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

## DETAILED ACTION

1.      Claims 44 and 45 have been cancelled.

2.      Claims 1-43 and 46-59 have been examined.

### *Response to Arguments*

3.      Applicant's arguments with respect to claims 1-43 and 46-59 have been considered but are

moot in view of the new ground(s) of rejection. See Moir et al. (US 2003/0174572 A1), new art

made of record.

### *Claim Rejections - 35 USC § 102*

4.      The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the

basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a
>
> foreign country, before the invention thereof by the applicant for a patent.

5.      Claims 1-15, 17-32, 34-43, 46-49 and 51-59 rejected under 35 U.S.C. 102(a) as being

anticipated by Moir et al. (US 2003/0174572 A1).

        Per claim 1 (Currently amended), Moir discloses a method ~~of providing non-blocking multi-~~

~~target transactions~~ in a computer system, the method comprising:

        defining a plurality of  transactionable locations, wherein individual ones of the

transactionable locations encode respective values (see at least paragraph [0029] "… multiprocessor

configuration … local storage portion… addressable memory, register, stack or other storage …")

and are owned by no more than one transaction at any given point in a multithreaded computation

(see at least paragraph [0012] "... synchronization constructs... transactional sequences..." and see

at least e.g. FIG. 1, P1, P2 and related text);

for a particular non-blocking multi-target transaction of the multithreaded computation

targeting two or more of the plurality of transactionable locations (see at least paragraph [0025] "...

non-blocking or which is stronger or weaker non-blocking properties.. lock free data structure...") ,

attempting to acquire ownership of each of the transactionable locations targeted thereby (see at

least paragraph [0047] "... thread passes a pointer... determine... is no longer in the shared data

structure... attempts to determine that the memory block is still in the data structure..."), wherein

the ownership acquiring wrests ownership from another non-blocking transaction, that owns the

targeted transactionable location without the other non-blocking transaction releasing ownership

(see at least paragraph [0036] "... synchronization facilities...", [0126] "... a single store of the

value to be guarded ..."); and

once ownership of each of the targeted transactionable locations has been acquired,

attempting to commit the particular non-blocking multi-target transaction using a single-target

synchronization primitive to ensure that, at the commit (see at least paragraph [0129] "... attempts

to had v off to g...", paragraph [0133], paragraph [0134] "... CAS attempts..." and FIG. 4 and

related text), the particular non-blocking multi-target transaction continues to own each of the

targeted transactionable locations, wherein individual ones of the non-blocking multi-target

transactions do not contribute to progress of another (see at least paragraph [0026] "... use of locks

... ensure that freed object are not subsequently accessed...", [0070] "... captures some important

properties of the support mechanism for non-blocking...", paragraph [0144] "... a value in its own

set...").

Per claim 2, Moir discloses the method of claim 1, wherein the ownership wresting employs

a single-target synchronization primitive to change status of the wrested from

transaction to be incompatible with a commit thereof (see at least paragraph [0036] "…

synchronization facilities…", [0126] "… a single store of the value to be guarded …").

Per claim 3, Moir discloses the method of claim 2, wherein, as a result of the status change,

the wrested from transaction fails and retries (see at least paragraph [0038] "… CAS fails due to

concurrent operation … second node in the list… retrying …").

Per claim 4 (Currently amended), Moir discloses the method of claim 2, wherein the wrested

from non-blocking transaction is itself a multi-target transaction (see at least e.g. FIG. 1, and related

text – multithreaded computation -).

Per claim 5, Moir discloses the method of claim 1, further comprising: on failure of the

commit attempt, reacquiring ownership of each targeted transactionable location and retrying (see

at least paragraph [0038] "… CAS fails due to concurrent operation … second node in the list…

retrying …").

Per claim 6, Moir discloses the method of claim 1, wherein no transaction may prevent

another from wresting therefrom ownership of transactionable locations targeted by the active

transaction (see at least paragraph [0119] "... deadlock node procedure uses a lock-free loop…").

Per claim 7, Moir discloses the method of claim 1, wherein the ownership acquiring employs

a single-target synchronization primitive to update the ownership of the targeted

transactionable location (see at least paragraph [0036] "… synchronization facilities…", [0126] "…

a single store of the value to be guarded …").

Per claim 8, Moir discloses the method of claim 1, wherein each encoding of a

transactionable location is atomically updateable using a single-target synchronization primitive (see

at least paragraph [0108] "… managing these version numbers… updated atomically with

pointers…").

Per claim 9, Moir discloses the method of claim 1, wherein the individual transactionable

location encodings further include an identification of the owning transaction's corresponding

value for the transactionable location (see at least paragraph [0113] "… stores a value v in the

location …").

Per claim 10, Moir discloses the method of claim 1, further comprising: accessing values

corresponding to individual ones of the transactionable locations using a wait-free load operation

(see at least paragraph [0128] "… remaining values… operation to be wait free…").

Per claim 11, Moir discloses the method of claim 1, wherein the transactionable locations

directly encode the respective values (see at least paragraph [0131] "… single location … for each

value v that has been passes…").

Per claim 12, Moir discloses the method of claim 1, wherein the transactionable locations are indirectly referenced (see at least paragraph [0030] "… local storage that references object in shared memory…").

Per claim 13, Moir discloses the method of claim 1, wherein the transactionable locations are encoded in storage managed using a non blocking memory management technique (see at least paragraph [0020] "… data structure… non blocking…").

Per claim 14, Moir discloses the method of claim 1, wherein the transactionable locations, if un owned, directly encode the respective values and otherwise encode a reference to the owning transaction (see at least paragraph [0128] "… remaining values… operation to be wait free…").

Per claim 15, Moir discloses the method of claim 1, wherein the single-target synchronization primitive employs a Compare-And-Swap (CAS) operation (see at least paragraph [0033] "… "… compare-and swap (CAS instruction)…").

Per claim 17, Moir discloses the method of claim 1, wherein the single-target of the single-target synchronization primitive includes at least a value and a transaction identifier encoded integrally therewith (see at least paragraph [0128] "… remaining values… operation to be wait free…").

Per claim 18 (Currently amended), Moir discloses the method of claim 1, wherein the <u>non-blocking</u> multi-target transaction has semantics of a multi-target compare and swap operation (see at least paragraph [0033] "… "… compare-and swap (CAS instruction)…").

Per claim 19, Moir discloses the method of claim 1, embodied in operation of an application programming interface (API) that includes a load operation and an multi-target compare and swap operation (see at least paragraph [0033] "… "… compare-and swap (CAS instruction)…").

Per claim 20, Moir discloses the method of claim 19, wherein the load operation is wait-free (see at least paragraph [0128] "… remaining values… operation to be wait free…").

Per claim 21, Moir discloses the method of claim 1, embodied in operation of an application programming interface (API) that provides transactional memory (see at least paragraph [0061] "… API for the guarding and liberating…").

Per claim 22 (Currently amended), Moir discloses a computer-readable storage medium program instructions computer-executable to implement:

a plurality of non-blocking, multi-target transactions (see at least paragraph [0029] "… multiprocessor configuration … local storage portion… addressable memory, register, stack or other storage …");

wherein the program instructions comprise:

instances of one or more single-target synchronization primitives executable to acquire, for a particular <u>non-blocking</u> multi-target transaction, ownership of <u>two or more</u> transactionable

locations <u>targeted by the non-blocking multi-target transaction;</u> (see at least paragraph [0012] "…

synchronization constructs… transactional sequences…" and see at least e.g. FIG. 1, P1, P2 and

related text) and

a particular single-target synchronization primitive executable to ensure that, at commit (see

at least paragraph [0129] "… attempts to had v off to g…", paragraph [0133], paragraph [0134] "…

CAS attempts…" and FIG. 4 and related text),

the particular <u>non-blocking</u> multi-target transaction continues to own each of the targeted

transactionable locations, wherein individual ones of the multi-target <u>non-blocking</u> transactions do

not contribute to progress of another (see at least paragraph [0025] "… non-blocking or which is

stronger or weaker non-blocking properties… lock free data structure…")


Per claim 23 (Currently amended), Moir discloses the storage medium claim 22, wherein the

program instructions are further executable to implement a concurrent computation, and wherein

execution of the concurrent computation invokes the multi-target <u>non-blocking</u> transactions (see at

least paragraph [0026] "… use of locks … ensure that freed object are not subsequently

accessed…", [0070] "… captures some important properties of the support mechanism for non-

blocking…", paragraph [0144] "… a value in its own set…").


Per claim 24 (Currently amended), Moir discloses the storage medium of claim 22, wherein

to acquire ownership, when performed by a first one of the <u>non-blocking</u> multi target transactions,

wrests ownership from respective other ones of the multi-target <u>non-blocking</u> transactions, that

own respective ones of the <u>two or more</u> transactionable locations (see at least paragraph [0119] "…

deadlock node procedure uses a lock-free loop…").

Per claim 25, Moir discloses the storage medium of claim 24, wherein to wrest ownership, the program instructions are further executable to implement an instance of a single-target synchronization primitive to change status of a wrested-from transaction to be incompatible with a commit thereof (see at least paragraph [0012] "… synchronization constructs… transactional sequences…" and see at least e.g. FIG. 1, P1, P2 and related text).

Per claim 26, Moir discloses  the storage medium of claim 25, wherein, as a result of the status change, the program instructions are further executable to complement the wrested-from transaction eventually fails and retries (see at least paragraph [0128] "… remaining values… operation to be wait free…").

Per claim 27 (Currently amended), Moir discloses the storage medium of claim 22, wherein no active transaction may prevent transaction another from wresting therefrom ownership of transactionable locations targeted by the active transaction (see at least paragraph [0038] "… CAS fails due to concurrent operation … second node in the list… retrying …").

Per claim 28 (Currently amended), Moir discloses the storage medium of claim 22, wherein the two or more transactionable locations directly encode the respective values (see at least paragraph [0038] "… CAS fails due to concurrent operation … second node in the list… retrying …").

Per claim 29 (Currently amended), Moir discloses the storage medium of claim 22, wherein

the transactionable locations are indirectly referenced (see at least paragraph [0131] "… single

location … for each value v that has been passes…").

Per claim 30 (Currently amended), Moir discloses the storage medium of claim 22, wherein

the two or more transactionable locations are encoded in storage managed using a non blocking

memory management technique (see at least paragraph [0020] "… data structure… non

blocking…").

Per claim 31 (Currently amended), Moir discloses the storage medium of claim 22, wherein

the two or more transactionable locations, if un owned, directly encode the respective values and

otherwise encode a reference to the owning transaction (see at least paragraph [0113] "… stores a

value v in the location …").

Per claim 32, Moir discloses the storage medium of claim 22, wherein at least some instances

of one or more the single-target synchronization primitive employ a Compare-And-Swap (CAS)

operation (see at least paragraph [0033] "… "… compare-and swap (CAS instruction)…").

Per claim 34 (Currently amended), Moir discloses the storage medium of claim 22, wherein

at least some of the multi-target non-blocking transaction have semantics of a multi-target compare

and swap operation (see at least paragraph [0033] "… "… compare-and swap (CAS

instruction)…").

Per claim 35 (Currently amended), Moir discloses the storage medium of claim 22, embodied

as software that includes a functional encoding of operations concurrently executable by one or

more processors to operate on state of the two or more transactionable locations (see at least

paragraph [0026] "… use of locks … ensure that freed object are not subsequently accessed…",

[0070] "… captures some important properties of the support mechanism for non-blocking…",

paragraph [0144] "… a value in its own set…").

Per claim 36 (Currently amended), Moir discloses the storage medium of claim 22, wherein

at least some of the multi-target non-blocking transactions are defined by an application

programming interface (API) that includes a load operation and a multi-target compare and swap

(NCAS) operation (see at least paragraph [0061] "… API for the guarding and liberating…").

Per claim 37 (Currently amended), Moir discloses the storage medium of claim 22, wherein

at least some of the multi-target non-blocking transactions are defined by an application

programming interface (API) that provides transactional memory (see at least paragraph [0061] "…

API for the guarding and liberating…").

Per claim 38 (Currently amended), Moir discloses storage medium of claim 22, wherein the

multi-target non-blocking transactions are obstruction-free, though not wait-free or lock-free (see

at least paragraph [0128] "… remaining values… operation to be wait free…").

Per claim 39 (Currently amended), Moir discloses the storage medium of claim 22, wherein

the implementation does not itself guarantee that at least one interfering concurrently executed

multi-target <u>non-blocking</u> transactions makes progress (see at least paragraph [0119] "... deadlock

node procedure uses a lock-free loop...").


Per claim 40, Moir discloses the storage medium of claim 22, wherein the program

instructions are further executable to implement a contention management facility configured to

facilitate progress in a concurrent computation (see at least paragraph [0119] "... deadlock node

procedure uses a lock-free loop...").


Per claim 41, Moir discloses the storage medium of claim 40, wherein operation of the

contention management facility ensures progress of the concurrent computation (see at least

paragraph [0026] "... use of locks ... ensure that freed object are not subsequently accessed...",

[0070] "... captures some important properties of the support mechanism for non-blocking...",

paragraph [0144] "... a value in its own set...").


Per claim 42, Moir discloses the storage medium of claim 40, wherein the contention

management facility is modular such that alternative contention management strategies may be

employed without affecting correctness of the implementation (see at least paragraph [0026] "...

use of locks ... ensure that freed object are not subsequently accessed...", [0070] "... captures

some important properties of the support mechanism for non-blocking...", paragraph [0144] "... a

value in its own set...").

Per claim 43, Moir discloses the storage medium of claim 40, wherein the contention

management facility allows changes in contention management strategy during a course of the

concurrent computation (see at least paragraph [0020] "... data structure... non blocking...").


Per claim 46 (Currently amended), Moir discloses a computer readable storage medium

storing program instructions computer-executable to implement (see at least paragraph [0119] "...

deadlock node procedure uses a lock-free loop..."):

instantiation of two or more transactionable locations in shared memory configured to

individually encapsulate values that may be targeted by concurrent executions of non-blocking multi-

target transactions (see at least paragraph [0029] "... multiprocessor configuration ... local storage

portion... addressable memory, register, stack or other storage ..."); and

one or more instances of a non-blocking multi-target transaction that upon execution of a

particular instance thereof, attempts to acquire ownership of each of a plurality of transactionable

locations targeted thereby and (see at least paragraph [0129] "... attempts to had v off to g...",

paragraph [0133], paragraph [0134] "... CAS attempts..." and FIG. 4 and related text), once

ownership of each of the targeted transactionable locations has been acquired, attempts to commit

the particular instance using a single-target synchronization primitive to ensure that, at the commit

(see at least paragraph [0036] "... synchronization facilities...", [0126] "... a single store of the value

to be guarded ..."), the particular instance continues to own each of the plurality of targeted

transactionable locations, wherein execution of no one of the non-blocking multi-target transaction

instances contributes to progress of another ((see at least paragraph [0026] "... use of locks ...

ensure that freed object are not subsequently accessed...", [0070] "... captures some important

properties of the support mechanism for non-blocking…", paragraph [0144] "… a value in its own set…").

Per claim 47, this is the computer readable medium version of the claimed method discussed above (Claim 7), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Moir.

Per claim 48, this is the computer readable medium version of the claimed method discussed above (Claim 4), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Moir.

Per claim 49, this is the computer readable medium version of the claimed method discussed above (Claim 15), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Moir.

Per claim 51, this is the computer readable medium version of the claimed method discussed above (Claim 17), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Moir.

Per claim 52, this is the computer readable medium version of the claimed method discussed above (Claim 1), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Moir.

Per claim 53, this is the computer readable medium version of the claimed method discussed above (Claim 18), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Moir.

Per claim 54, this is the computer readable medium version of the claimed method discussed above (Claim 5), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Moir.

Per claim 55, Moir discloses a encoding of claim 46, wherein the computer readable medium includes at least one medium selected from the set of a disk, tape or other magnetic, optical, or electronic storage medium and a network, wire line, wireless or other communications medium (e.g. FIG. 1 and related text).

Per claim 56, this is the apparatus version of the claimed method discussed above (Claim 1), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Moir.

Per claim 57, this is the computer readable medium version of the claimed method discussed above (Claim 7), wherein all claim limitations have been addressed and/or covered in cited areas as set forth above. Thus, accordingly, these claims are also anticipated by Moir.

Per claim 58, this is the computer readable medium version of the claimed method discussed

above (Claim 2), wherein all claim limitations have been addressed and/or covered in cited areas as

set forth above. Thus, accordingly, these claims are also anticipated by Moir.

Per claim 59, this is the computer readable medium version of the claimed method discussed

above (Claim 4), wherein all claim limitations have been addressed and/or covered in cited areas as

set forth above. Thus, accordingly, these claims are also anticipated by Moir.

### *Claim Rejections - 35 USC § 103*

6.      The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness

rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section
>
> 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the
>
> subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary
>
> skill in the art to which said subject matter pertains.  Patentability shall not be negatived by the manner in which the
>
> invention was made.

7.      Claims 16, 33 and 50 rejected under 35 U.S.C. 103(a) as being unpatentable over Moir in

view of Maged et al ("Non-Blocking Algorithms and Preemption-Safe Locking on

Multiprogrammed Shared Memory Multiprocessors", March 1997).

Per claims 16, 33 and 50 Moir do not explicitly disclose wherein the single-target

synchronization primitive employs Load-Linked (LL) and Store-Conditional (SC) operation pair.

However, Maged et al. discloses non blocking algorithms (See Section 3, page 5). Figure 2, page 7

shows a non-blocking counter implementation using load-linked/store –conditional. In addition,

Maged et al. discloses emulation using load-linked and store – conditional instruction (page 13).

Therefore it would have been obvious to employ Load-linked and Store-Conditional operation to

read, modify and write a shared location as once suggested by Maged et al. (page 3).

## Conclusion

8.      Applicant's amendment necessitated the new ground(s) of rejection presented in this Office

action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is

reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS

from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the

mailing date of this final action and the advisory action is not mailed until after the end of the

THREE-MONTH shortened statutory period, then the shortened statutory period will expire on

the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be

calculated from the mailing date of the advisory action. In no event, however, will the statutory

period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner

should be directed to ISAAC T. TECKLU whose telephone number is (571) 272-7957. The

examiner can normally be reached on M-TH 9:300A - 8:00P.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor,

Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization

where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR system,

see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system,

contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like

assistance from a USPTO Customer Service Representative or access to the automated information

system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.


/Isaac T Tecklu/                                          /Tuan Q. Dam/
Examiner, Art Unit 2192                                   Supervisory Patent Examiner, Art Unit 2192